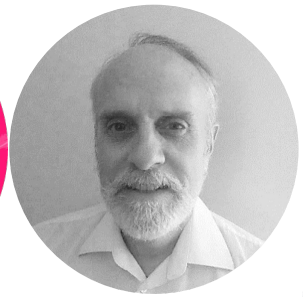
The Strategy Pattern



Gerald Britton

Pluralsight Author

@GeraldBritton www.linkedin.com/in/geraldbritton

Overview

Classification: Behavioral

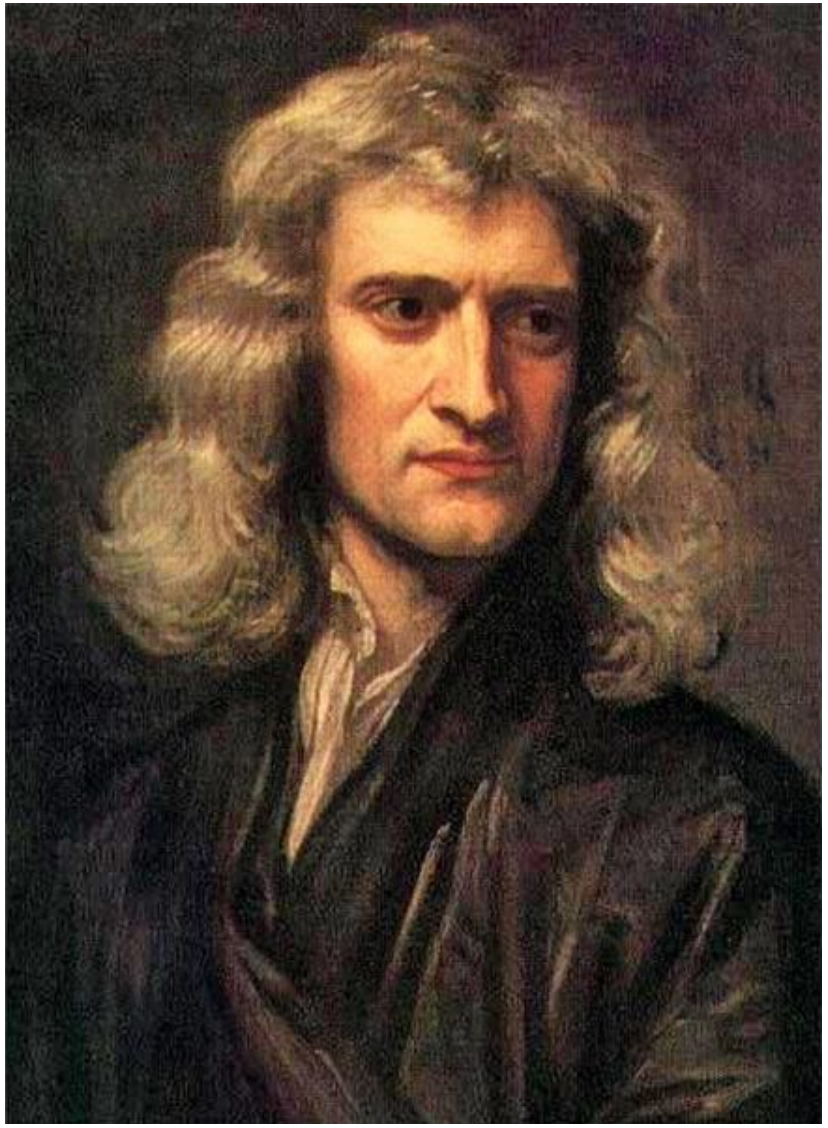
Family of algorithms

Encapsulate each one

Make them interchangeable

Algorithms vary independently

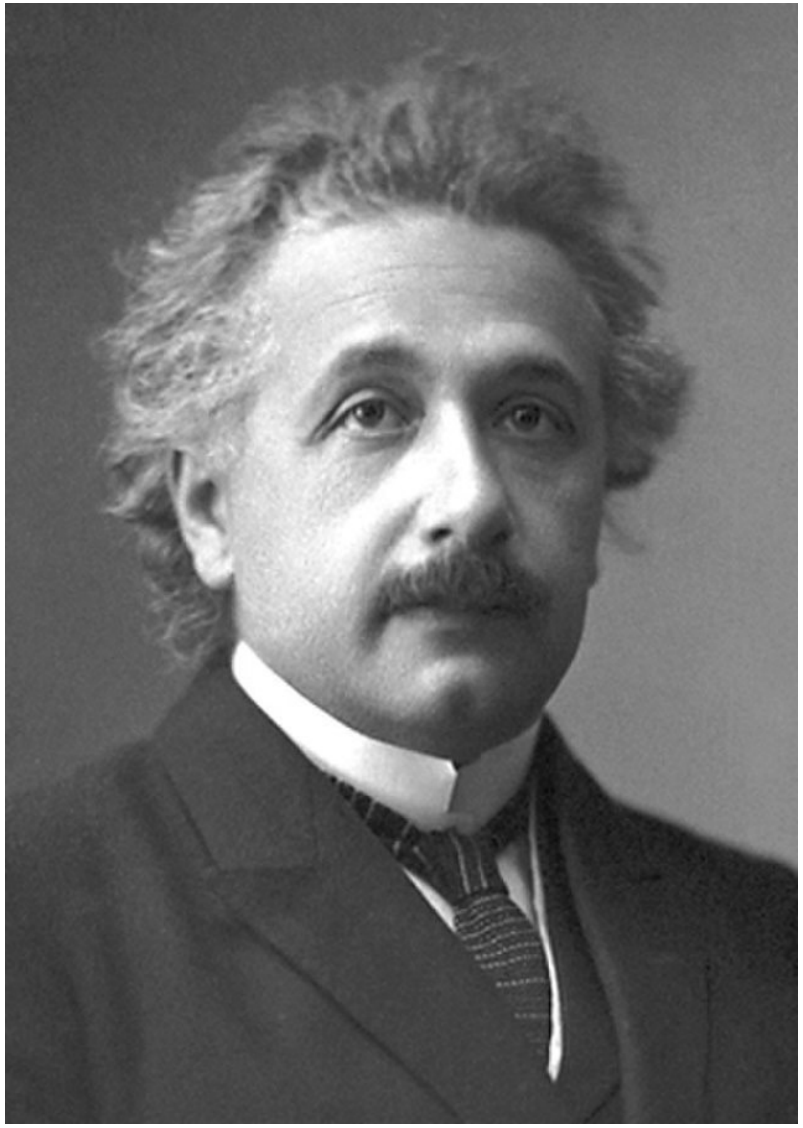
Also know as the Policy pattern



Godfrey Kneller, Public domain, via Wikimedia Commons

$$F = G\frac{m_1 m_2}{r^2}$$

$$G_{\mu\nu}+\Lambda g_{\mu\nu}= \frac{8\pi G}{c^4}T_{\mu\nu}$$



Nobel foundation / A.B. Lagrelius & Westphal, Public domain, via Wikimedia Commons

Demo

Motivating example

Shipping cost calculator

Must support:

- Federal Express
- UPS
- Postal Service

Must be extendable (add new shippers)

Problems Discovered

Violates Single Responsibility Principle

Violates Open/Closed Principle

Violates Dependency Inversion Principle

Long list of `if/elif` clauses

Strategy Pattern Structure

ShippingCost Strategy Structure

Demo

Fix the problems discovered

Remove shipping concerns from orders

Separate the algorithms

Keep the classes closed

Allow for extension

Program to an interface (Python ABC)

Advantages of the Strategy Pattern

Fixed the problems we discovered

Test algorithms in isolation

Test the outer code with deterministic mock algorithms

No more `if/elif/else` statement

Demo

Variations:

- Strategies as functions
- Strategies as lambdas

Summary

Encapsulate algorithms

Several techniques available

- Class per algorithm
- Function definitions
- Lambda expressions

Sequences of `if/elif/else` are a red flag